Seminar Software Quality

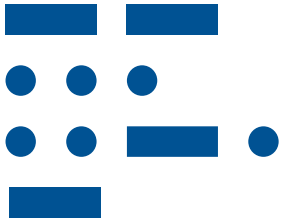
Preliminary meeting

We will start at 13:32

Fabian Leinen (Orga)

Jakob Rott
Roland Würsching
Dr. Markus Schnappinger
Maximilian Jungwirth
Dr. Martin Gruber
Xin Ye
Roman Haas

Software Quality



Code

Models

Tests

Participating

1. Apply via matching tool

2. Application with us: Online form
   - Letter of motivation
   - Optional: CV + grade report
   - Your 3+ favorite topics

http://go.tum.de/070420

July 22nd, 23:59

Schedule

- Literature research
- How to thesis?
- Effective presentations

Application

Matching

Matching results and topic assignment

Kickoff

Individual working phase

Block seminar

Until 22nd of July — **October** — **February**

Grading

Thesis

- Seminar paper: max. 15 pages
- Content: Theory + **application** of the topic
  (results, experiences, problems and limitations)
- Initial submission
- Final submission: 1 week after presentation

Presentation

- 20 min + 10 min discussion
- Mandatory dry run (1 week before seminar)

50/50

Questions about the organization?

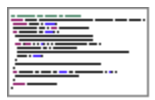
Clone Detection:

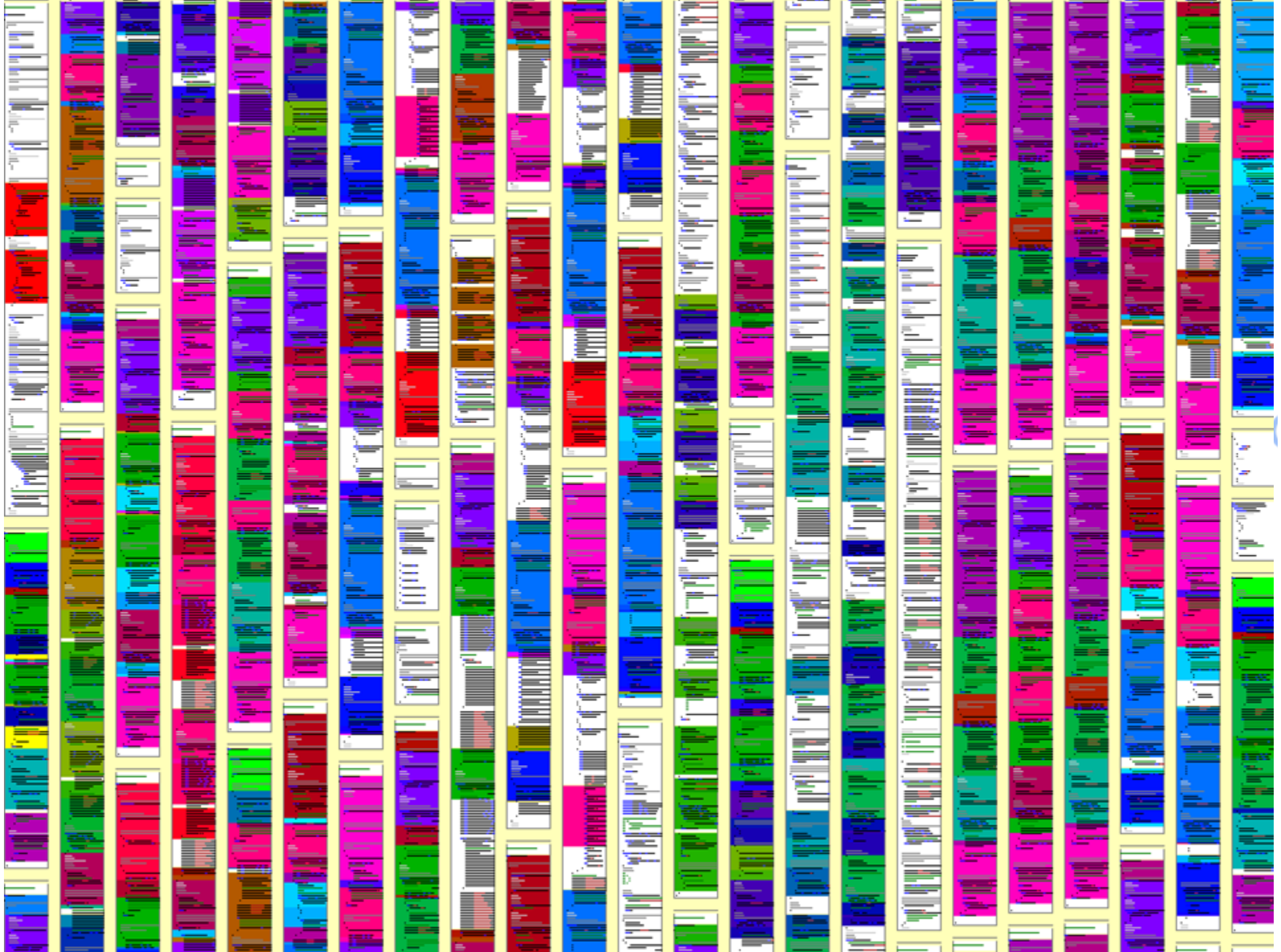
"Where can identical (copied) parts be found in source code?”

```java
// Utilities for arrays of elements
public String showElements(ModelElement[] elements, String nomsg) {
    boolean found = false;
    StringBuffer res = new StringBuffer();
    if (elements != null) {
        Index.getInstance().setCurrentRenderer(
                FlatReferenceRenderer.getInstance());
        for (int i = 0; i < elements.length; i++) {
            ModelElement el = elements[i];
            res.append(showElementLink(el)).append(HTML.LINE_BREAK);
            found = true;
        }
        Index.getInstance().resetCurrentRenderer();
    }
    if (!found && nomsg != null && nomsg.length() > 0) {
        res.append(HTML.italics(nomsg));
    }
    return res.toString();
}
```

```java
// Utilities for arrays of elements
public String showElements(ModelElement[] elements, String nomsg) {
    boolean found = false;
    StringBuffer res = new StringBuffer();
    if (elements != null) {
        Index.getInstance().setCurrentRenderer(
                FlatReferenceRenderer.getInstance());
        for (int i = 0; i < elements.length; i++) {
            ModelElement el = elements[i];
            res.append(showElementLink(el)).append(HTML.LINE_BREAK);
            found = true;
        }
        Index.getInstance().resetCurrentRenderer();
    }
    if (!found && nomsg.length() > 0) {
        res.append(HTML.italics(nomsg));
    }
    return res.toString();
}
```

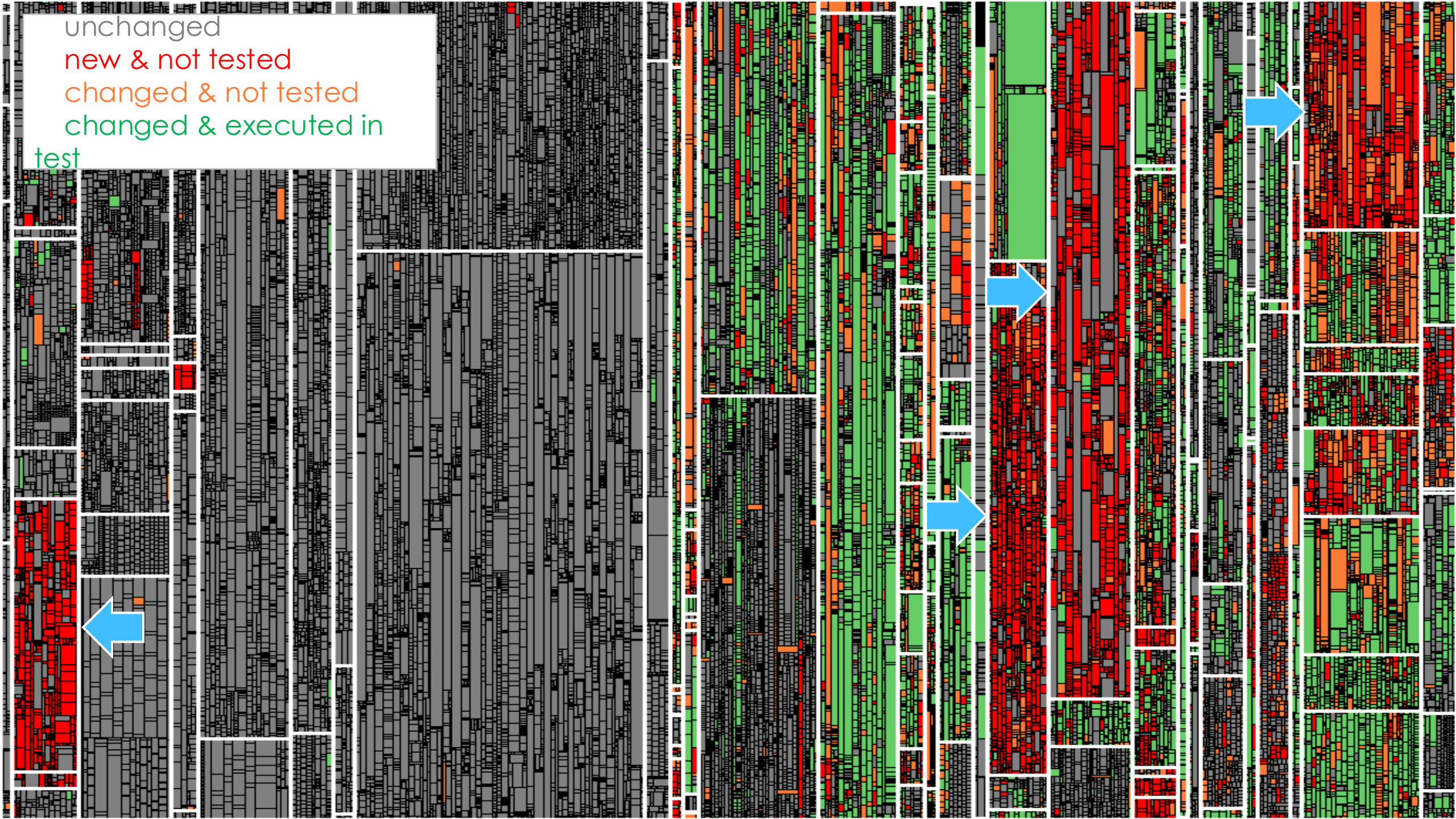
```java
// Utilities for arrays of elements
public String showElements(ModelElement[] elements, String nomsg) {
    boolean found = false;
    StringBuffer res = new StringBuffer();
    if (elements != null) {
        Index.getInstance().setCurrentRenderer(
                FlatReferenceRenderer.getInstance());
        for (int i = 0; i < elements.length; i++) {
            ModelElement el = elements[i];
            res.append(showElementLink(el)).append(HTML.LINE_BREAK);
            found = true;
        }
        Index.getInstance().resetCurrentRenderer();
    }
    if (!found && nomsg != null && nomsg.length() > 0) {
        res.append(HTML.italics(nomsg));
    }
    return res.toString();
}
```





Test Gap Analysis

"Have all changes since the last release been tested?"


unchanged
new & not tested
changed & not tested
changed & executed in test

Project		Final Score	Quality Score	Delta Score	Source Lines of Code	Longest Method Length	Method Length Assessment	Nesting Depth Assessment	Findings Density	Clone Coverage
	GREEN	0.7340755143920171	0.5077306289966398	0.8095238095238095	0.42857142857142855	0.6666666666666666	0.6309523809523809	0.30952380952380953	0.5833333333333334	0.8333333333333334
	GREEN	0.6621189876776317	0.21990452213909772	0.8095238095238095	0.25	0.047619047619047616	0.4166666666666667	0.23809523809523808	0.38095238095238093	0.2857142857142857
	GREEN	0.5361766327836325	0.6804208168488156	0.4880952380952381	0.5714285714285714	0.44047619047619047	0.5714285714285714	1	0.9880952380952381	0.75
	GREEN	0.6648463747493384	0.23081407042592458	0.8095238095238095	0.4166666666666667	0.2976190476190476	0.13095238095238096	0.13095238095238096	0.20238095238095238	0.2261904761904762
	YELLOW	0.3473775609928538	0.3180816725428438	0.35714285714285715	0.32142857142857145	0.4166666666666667	0.5	0.15476190476190477	0.36904761904761907	0.40476190476190477
	YELLOW	0.39419802613174176	0.2553635330983956	0.44047619047619047	0.05952380952380952	0.4880952380952381	0.6190476190476191	0.3333333333333333	0.39285714285714285	0.5357142857142857
	GREEN	0.8236932149244094	0.2947728596976374	1	0.14285714285714285	0.23809523809523808	0.4880952380952381	0.6309523809523809	0.30952380952380953	0.32142857142857145
	GREEN	0.5437473139473112	0.6749892557892448	0.5	0.6190476190476191	0.8333333333333334	0.8214285714285714	0.8333333333333334	0.44047619047619047	0.5595238095238095
	GREEN	0.7874932261864153	0.22140147617423267	0.9761904761904762	0.6309523809523809	0.34523809523809523	0.32142857142857145	0.09523809523809523	0.03571428571428571	0.19047619047619047
	GREEN	0.6336629973650774	0.10608056088888057	0.8095238095238095	0.2261904761904762	0.4642857142857143	0.16666666666666666	0.03571428571428571	0.05952380952380952	0.023809523809523808
	GREEN	0.6454584818857084	0.15326249897140481	0.8095238095238095	0.7857142857142857	0.07142857142857142	0.10714285714285714	0.023809523809523808	0.2261904761904762	0.20238095238095238
	RED	0.1606877082158262	0.5713222614347334	0.023809523809523808	0.023809523809523808	0.19047619047619047	0.6666666666666666	0.7976190476190477	0.7380952380952381	0.38095238095238093
	RED	0.16961627964439763	0.5713222614347334	0.03571428571428571	0.35714285714285715	0.6071428571428571	0.8809523809523809	0.6547619047619048	0.8571428571428571	0.5476190476190477
	RED	0.15503738573196005	0.2987209714992688	0.10714285714285714	0.11904761904761904	0.30952380952380953	0.40476190476190477	0.36904761904761907	0.5714285714285714	0.6190476190476191
	RED	0.07363249735450234	0.15167284656086646	0.047619047619047616	0.39285714285714285	0.5238095238095238	0.36904761904761907	0.05952380952380952	0.023809523809523808	0.047619047619047616
	YELLOW	0.36510741763059196	0.567572527665225	0.2976190476190476	0.5952380952380952	0.25	0.42857142857142855	0.5833333333333334	0.9761904761904762	1
	YELLOW	0.2619985765122256	0.4408514489060451	0.20238095238095238	0.7976190476190477	0.5714285714285714	0.19047619047619047	0.14285714285714285	0.5476190476190477	0.9642857142857143
	GREEN	0.752153058971213	0.5800408073134232	0.8095238095238095	0.47619047619047616	0.8452380952380952	0.7619047619047619	0.47619047619047616	0.6071428571428571	0.5833333333333334
	YELLOW	0.19282558282740156	0.12844518845246342	0.21428571428571427	0.21428571428571427	0.05952380952380952	0.047619047619047616	0.08333333333333333	0.6666666666666666	0.11904761904761904
	YELLOW	0.38873567894641003	0.3406570014999257	0.40476190476190477	0.8690476190476191	0.9166666666666666	0.44047619047619047	0.19047619047619047	0.15476190476190477	0.05952380952380952

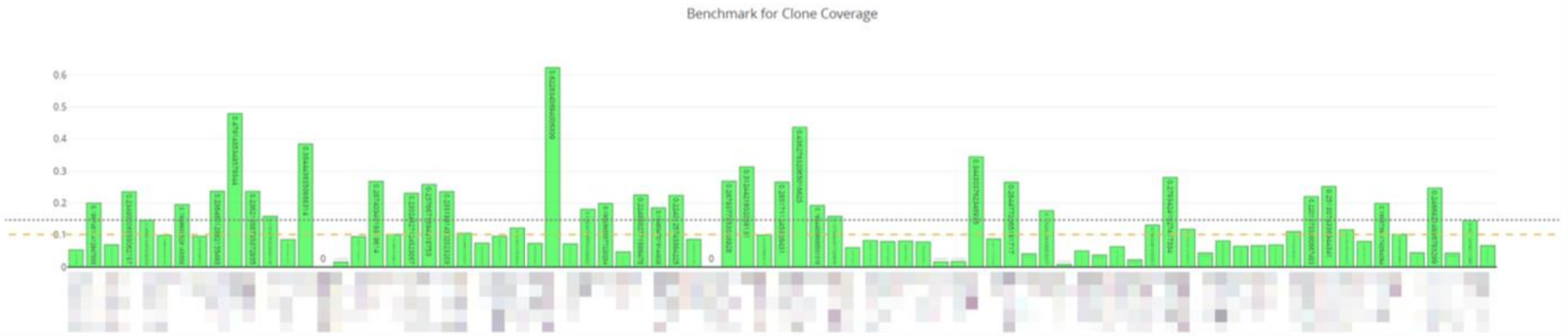
Overview | Advanced

Aggregated Metrics

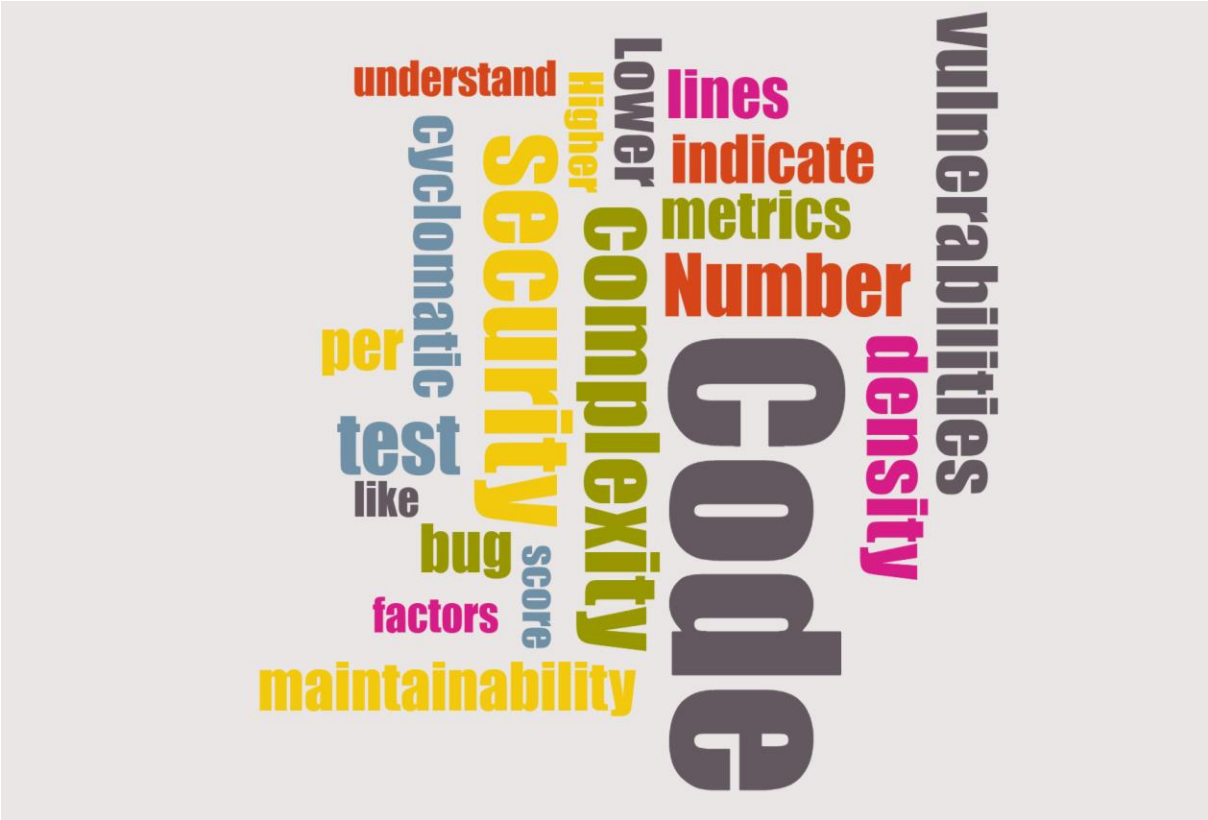
Source Lines of Code × | Longest Method Length × | Method Length Assessment × | Nesting Depth Assessment × | Findings Density × | Clone Coverage ×

Metric Benchmark

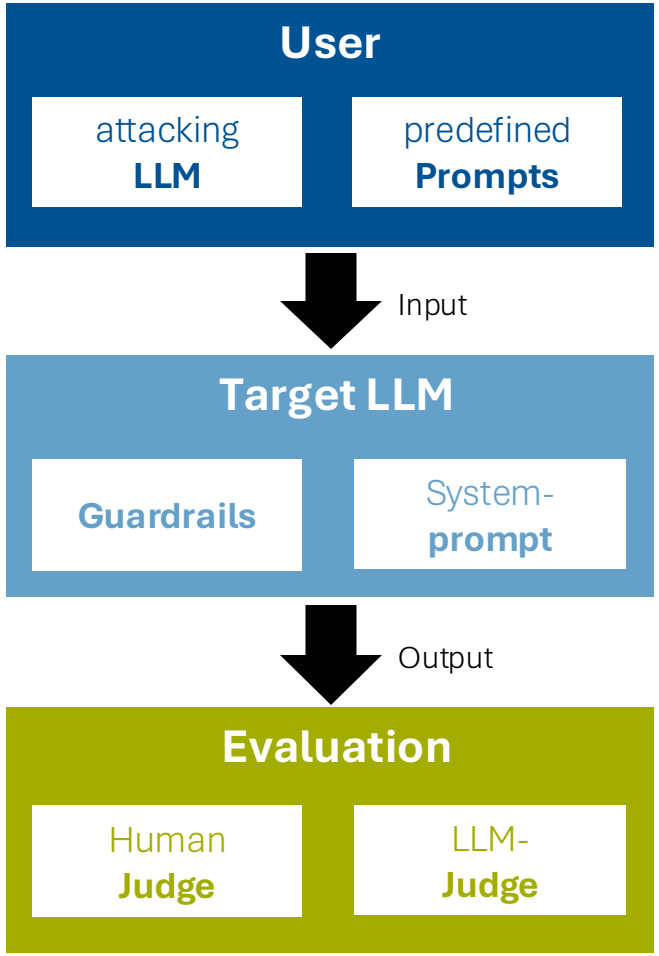
Clone Coverage



Why and How to Measure Code Quality?



A Pipeline for Evaluating Guardrails for LLMs



Your seminar will focus on:

- researching the state-of-the-art considering guardrails and system prompts
- developing a **technical pipeline** to automatically check if an LLM adheres to its defined guardrails and follows its system prompts. To check if the instructions were violated, both humans or other LLMs may act as judges, as well as an ensemble of LLMs
- using the pipeline as the experimental platform **to analyze and compare** different guardrail techniques and their corresponding attack scenarios
- We will refine the exact research questions together based on your interests

Looking forward to working with you!

Quality Issues In Natural Language Tests

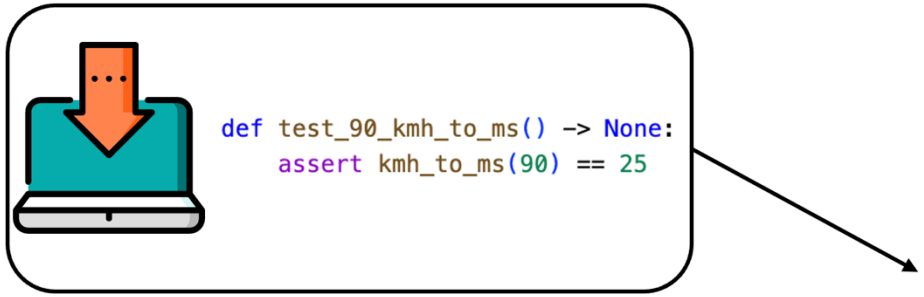
Step	Description	Expected Result
1	Launch the browser.	Browser is started.
2	Click menu → select 'Customize'.	The 'Customize' window is opened.
3	Drag 3 new items from the palette or menu panel and drop them onto the Navigation toolbar.	All items are added onto the Navigation toolbar.
4	Exit 'Customize'.	The changes are applied.
5	Wait at least 15 seconds, after exiting 'Customize', then restart the browser.	Browser is restarted and the previously made customizations are in place.

(1) **Identification** of quality issues

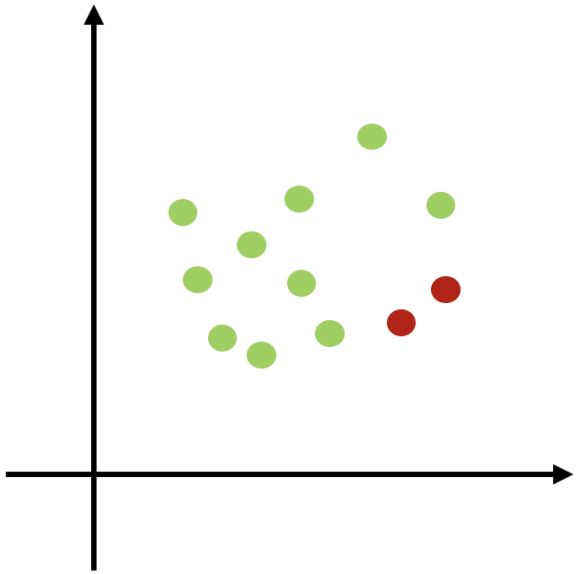
- Ambiguous descriptions
- Long test steps
- Misplaced actions
- Inconsistent wording
- etc.

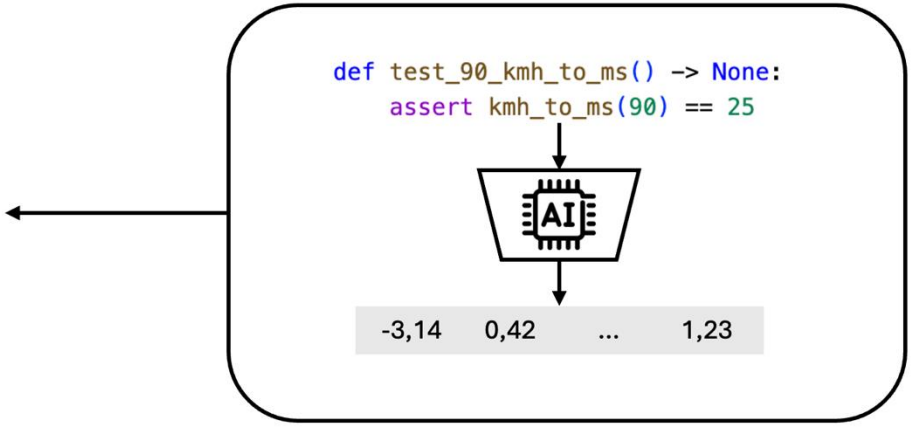
(2) Automated **improvement**

Using Pre-Trained Embedding Models for Diversity-Based Regression Test Optimization



```python
def test_90_kmh_to_ms() -> None:
    assert kmh_to_ms(90) == 25
```

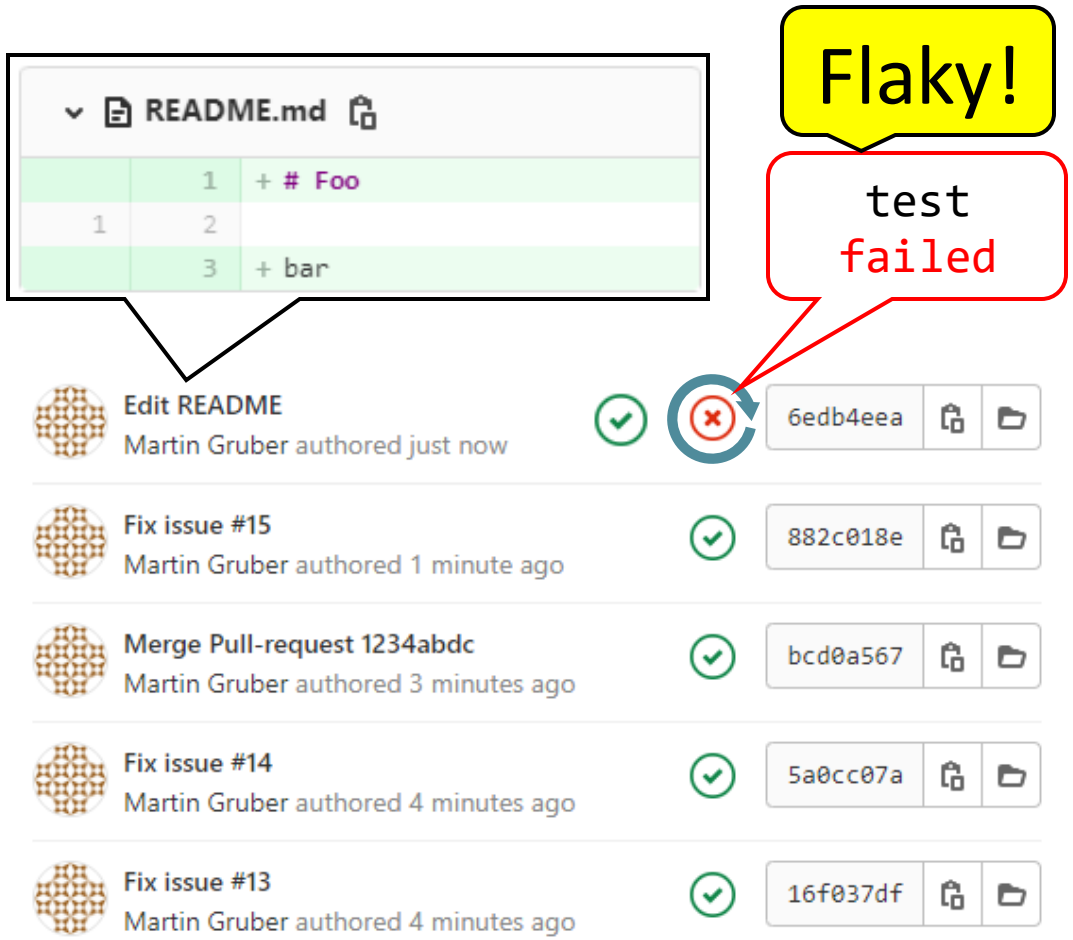


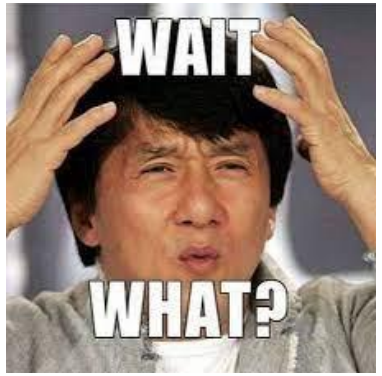


JaCoCo

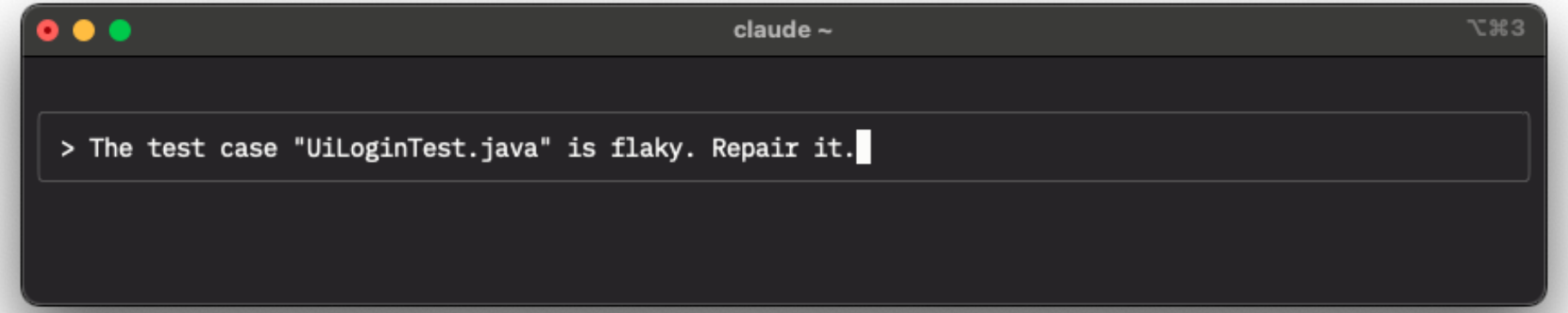
Element	Missed Instructions	Cov.	Missed Branches	Cov.
org.jacoco.core		97%		90%
org.jacoco.examples		58%		64%
org.jacoco.agent.rt		75%		83%
jacoco-maven-plugin		90%		82%
org.jacoco.cli		97%		100%
org.jacoco.report		99%		99%
org.jacoco.ant		98%		99%
org.jacoco.agent		86%		75%
Total	1,454 of 29,386	95%	206 of 2,442	91%

The Origin of Test Flakiness





Flaky Test, Solid Solution?
Repairing Flaky Tests Using LLMs



Problem:
The LLM will make changes. Are these changes *good*?
- Does the change effectively repair the flakiness?
- Is the test case still capable of finding regressions?

Potential solutions:
- Extend an approach that *finds* flaky tests by your automated repair approach
- Use personal projects or from your working position to try out your approach
- Benchmark different products (Claude Code, Cursor, Gemini CLI, ...)

Evaluating LLM-Generated Test Cases: The Core Problem & Framework

The use of Large Language Models to generate test cases is rapidly increasing, but our ability to evaluate and trust these tests has not kept pace

We will be looking together creating an approach that support the following aspects

1. Correctness
2. Completeness
3. Quality

Participating

1. Apply via matching tool

2. Application with us: Online form
   - Letter of motivation
   - Optional: CV + grade report
   - Your 3+ favorite topics

http://go.tum.de/070420

July 22nd, 23:59